

```
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
public class LibraryCatalogSystem
{
    static class Book
    {
        private String title;
        private String author;
        private String isbn;
        private boolean isAvailable;
        public Book(String title, String author, String isbn)
        {
            this.title = title;
            this.author = author;
            this.isbn = isbn;
            this.isAvailable = true;
        }

        public String getTitle()
        {
            return title;
        }

        public String getAuthor()
        {
            return author;
        }

        public String getIsbn()
        {
            return isbn;
        }

        public boolean isAvailable()
        {
            return isAvailable;
        }

        public void setAvailable(boolean available)
        {
            isAvailable = available;
        }
    }
    static class LibraryCatalog
    {
        private final List<Book> books;
        public LibraryCatalog()
        {
            this.books = new ArrayList<>();
```

```
        }

    public void addBook(Book book)
    {
        for (Book b : books)
        {
            if (b.getIsbn().equals(book.getIsbn()))
            {
                System.out.println("\n[ERROR] Book with ISBN " + book.getIsbn()
+ " already exists.");
                return;
            }
        }
        books.add(book);
        System.out.println("\n[SUCCESS] Added book: " + book.getTitle());
    }

    public List<Book> searchByTitle(String title)
    {
        List<Book> results = new ArrayList<>();
        String lowerCaseTitle = title.toLowerCase();
        for (Book book : books)
        {
            if (book.getTitle().toLowerCase().contains(lowerCaseTitle))
            {
                results.add(book);
            }
        }
        return results;
    }

    public List<Book> searchByAuthor(String author)
    {
        List<Book> results = new ArrayList<>();
        String lowerCaseAuthor = author.toLowerCase();
        for (Book book : books)
        {
            if (book.getAuthor().toLowerCase().contains(lowerCaseAuthor))
            {
                results.add(book);
            }
        }
        return results;
    }

    public void listAllBooks()
    {
        if (books.isEmpty())
        {
            System.out.println("The catalog is currently empty.");
        }
    }
}
```

```
        return;
    }
    System.out.println("\n Listing All Books (" + books.size() + " total)");
    for (int i = 0; i < books.size(); i++)
    {
        Book book = books.get(i);
        System.out.println("Book #" + (i + 1) + ":");
        System.out.println(" Title: " + book.getTitle());
        System.out.println(" Author: " + book.getAuthor());
        System.out.println(" ISBN: " + book.getIsbn());
        System.out.println(" Status: " + (book.isAvailable() ? "Available" :
"Checked Out"));
    }
}
public static void main(String[] args)
{
    try (Scanner scanner = new Scanner(System.in))
    {
        LibraryCatalog catalog = new LibraryCatalog();
        boolean running = true;
        System.out.println("LIBRARY CATALOG SYSTEM");
        while (running)
        {
            System.out.println("\nChoose an option:");
            System.out.println("1. Add New Book");
            System.out.println("2. Search Book by Title");
            System.out.println("3. Search Book by Author");
            System.out.println("4. List All Books");
            System.out.println("5. Exit System");
            System.out.print("Enter choice (1-5): ");
            try
            {
                int choice = scanner.nextInt();
                scanner.nextLine();
                switch (choice)
                {
                    case 1:
                        System.out.println("ADD NEW BOOK");
                        System.out.print("Enter Title:");
                        String title = scanner.nextLine().trim();
                        System.out.print("Enter Author:");
                        String author = scanner.nextLine().trim();
                        System.out.print("Enter ISBN:");
                        String isbn = scanner.nextLine().trim();
                        Book newBook = new Book(title, author, isbn);
                        catalog.addBook(newBook);
                        break;
                    case 2:
                        System.out.println("SEARCH BY TITLE");
                }
            }
            catch (InputMismatchException e)
            {
                System.out.println("Invalid choice. Please enter a number between 1 and 5.");
            }
        }
    }
}
```

```
        System.out.print("Enter Title search term:");
        String searchTitle = scanner.nextLine().trim();
        List<Book> titleResults =
catalog.searchByTitle(searchTitle);
        displayResults(titleResults, "Title");
        break;
    case 3:
        System.out.println("SEARCH BY AUTHOR");
        System.out.print("Enter Author search term:");
        String searchAuthor = scanner.nextLine().trim();
        List<Book> authorResults =
catalog.searchByAuthor(searchAuthor);
        displayResults(authorResults, "Author");
        break;
    case 4:
        catalog.listAllBooks();
        break;
    case 5:
        System.out.println("Thank you for using the Library
Catalog System. Goodbye!");
        running = false;
        break;
    default:
        System.out.println("[ERROR] Invalid choice. Please enter
a number between 1 and 5.");
    }
}
catch (java.util.InputMismatchException e)
{
    System.out.println("\n[ERROR] Invalid input. Please enter a
numerical choice.");
    scanner.nextLine();
}
}
catch (Exception e)
{
    System.out.println("An unexpected error occurred: " + e.getMessage());
}
}

private static void displayResults(List<Book> results, String searchType)
{
    if (results.isEmpty())
    {
        System.out.println("No books found matching the " + searchType + " search
term.");
        return;
    }
    System.out.println("Found " + results.size() + " book(s) matching your
search.");
}
```

```
        Book book = results.get(i);
        System.out.println("Result #" + (i + 1) + ":");
        System.out.println(" Title: " + book.getTitle());
        System.out.println(" Author: " + book.getAuthor());
        System.out.println(" ISBN: " + book.getIsbn());
        System.out.println(" Status: " + (book.isAvailable() ? "Available" :
"Checked Out"));
    }
}
```